Published in *Proceedings of the 45th IEEE International Midwest Symposium on Circuits and Systems,* Tulsa, Oklahoma, Aug 2002.

On-line, gyro-based, mass-property identification for thruster-controlled spacecraft using recursive least squares[1]

Edward Wilson[2], Chris Lages[2], and Robert Mah[3]
Smart Systems Research Laboratory, NASA Ames Research Center, MS 269-1, Moffett Field, CA 94035

Abstract

Spacecraft control, state estimation, and fault-detection-and-isolation systems are affected by unknown variations in the vehicle mass properties. It is often difficult to accurately measure inertia terms on the ground, and mass properties can change on-orbit as fuel is expended, the configuration changes, or payloads are added or removed. Recursive least squares-based algorithms that use gyro signals to identify the center of mass and inverse inertia matrix are presented. They are applied in simulation to 3 thruster-controlled vehicles: the X-38 and Mini-AERCam under development at NASA-JSC, and the S4, an air-bearing spacecraft simulator at the NASA-Ames Smart Systems Research Lab (SSRL).

1. Introduction

The mass-property identification (ID) algorithms presented here were developed through application to two thruster-controlled spacecraft presently under development at NASA Johnson Space Center: the X-38 [14] and the Mini-AERCam, and to the SSRL S4.

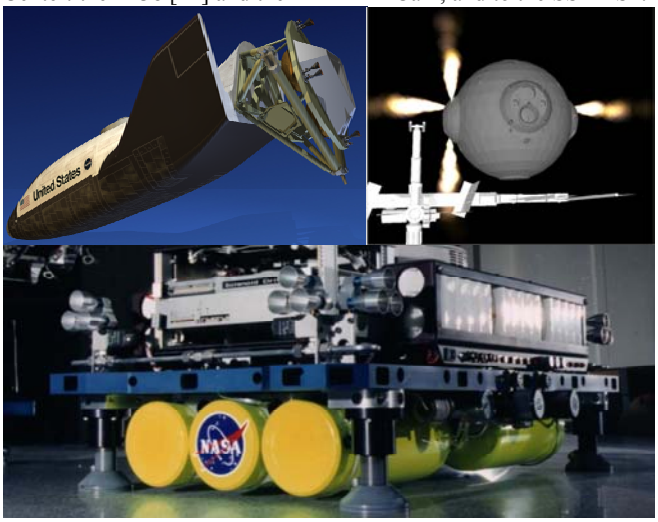

Fig 1: X-38 (upper left), Mini-AERCam (upper right), and Smart Systems Spacecraft Simulator (S4)

The goal is to identify, on-line, the mass property parameters of a thruster-controlled spacecraft. Specifically, the terms of the inverse inertia matrix and the location of the mass center are identified using gyro measurements during periods of thruster firing. The application vehicles, X-38, Mini-AERCam, and S4 are all exclusively thruster controlled and the derivation follows that, but it is a simple extension to accommodate CMGs or reaction wheels (as in [2]). Accelerometers would improve the ID, but gyros only are used, broadening the applicability of the resulting algorithm.

[1] Research funded by NASA Headquarters, HQ AA, PWC 349-00.
[2] Drs. Wilson and Lages are with Intellization, ed.wilson and chris.lages@intellization.com
[3] Dr. Mah is with NASA Ames Research Center, rmah@mail.arc.nasa.gov

1.1 Related research

Tanygin and Williams [9] developed a least squares (LS) based algorithm to identify mass properties for a spinning vehicle during coasting maneuvers. Bergmann, *et al.* [2][3][8] developed an ID approach using a Gaussian second-order filter [5], which resembles an extended Kalman filter. This is significantly more complex and computationally intensive than the approach presented here. Wilson and Rock [11][12] developed an ID method based on exponentially weighted RLS using accelerometer and angular rate sensors. The acceleration created by each thruster (reflecting both mass and thruster properties) was identified. A neural network then provided adaptive control reconfiguration to multiple destabilizing hard and soft thruster failures. This was implemented on a 3-degree-of-freedom air-bearing vehicle. Wilson, *et al.* developed a model-based thruster fault detection and isolation (FDI) system for the vehicles presented here [13]. These mass-property ID algorithms were developed to improve model accuracy for that system.

1.2 Least-squares identification

In this paper, vehicle mass properties are identified using LS methods in which the sensor data is fit to the underlying equations of motion (EOM) such that the identified parameter values minimize the squared error (where error is, for example, sensor data minus the ideal sensor data that would occur with zero noise and using the identified parameter values).

The standard form for a linear least squares problem is given as

$$Ax = b + \varepsilon \quad \text{or} \quad Ax \cong b$$

where b is a vector of noise-free measurements, ε is a vector of measurement noise, x contains the parameters to be identified, and matrix A contains known variables and parameters (*i.e.*, A is noise-free). The $\cong$ in $Ax \cong b$ indicates that the left and right sides of the equation would be equal if noise were not present [6]. The LS ID solution, $\hat{x}$, minimizes the sum of the squares of the error, $A\hat{x} - b$. If the problem at hand can be put into this standard form, $\hat{x}$ can be solved directly using a batch algorithm, $\hat{x} = (A^T A)^{-1} A^T b$, or an exactly equivalent recursive algorithm [4][5][6]. Manipulating the original equations into the form $Ax \cong b$ so that the standard LS solution can be solved is often the primary challenge, requiring careful, application-dependent decisions regarding approximations.

The rotational EOM (derived later) contain all of the parameters to be identified: center of mass (CM), as represented in thruster location in the body frame, L, and inertia. Unfortunately, these parameters multiply one another, and cannot *all* simultaneously be manipulated into the desired linear form, $Ax \cong b$. The approach taken here addresses this issue by segmenting the ID problem into two sub-problems which both allow closed form solution. Implementing both recursively, and with each ID solution using the results of the other one (with confidence checks), the ID bias due to coupling is minimized.

2. Approach

For the vehicle applications studied, the presence of disturbance torques, imperfect rate measurements, and significant pulse-to-pulse thruster variability biased development towards one of minimal complexity (*e.g.*, dropping higher order terms, segmenting the problem, *etc.*). Before going into the equations, it is useful to understand the basic physics behind the ID approach:

1. The rotational EOM mathematically describe how forces and torques will affect rotational accelerations. The inertia matrix governs the acceleration that will result from an applied net torque, allowing LS ID of I^{-1}. The CM location determines the angular acceleration that will result from an applied force.
2. When thrusters are fired to produce a pure torque only, the CM location has no effect on angular acceleration. So the CM ID cannot be updated using this data. Also, inertia ID will not be biased by an incorrect CM ID estimate.
3. When force and torque are applied simultaneously, inertia and CM can each be identified *individually*.
4. When no external forces or torques are applied, disturbances due to thruster variability are removed. With sufficiently high angular rates and sufficiently low sensor noise, accelerometer signals can be used along with the gyros to ID the CM location. This has been developed and implemented by the authors for a vehicle with relatively low sensor noise levels (and in [9]), and is not presented here.

Specifics regarding this approach for gyro-based mass-property ID:

- Due to the form of the EOM, the inverse of the inertia matrix is identified (*i.e.*, $\hat{I}^{-1}$ *vs.* $\hat{I}$).
- Only the six independent elements of $\hat{I}^{-1}$ are identified.
- $\hat{I}^{-1}$ and CM updates are made only when thrusters are firing.
- Perfect knowledge of thruster failures and biases is assumed.
- If pure torques occur frequently, item 2 above is followed, so $\hat{I}^{-1}$ updates are made *only* when a pure torque is present (assuming that fewer clean data points are better than more noisy ones, and that the CM error would bias the estimate).

3. Derivation of mass-property RLS ID algorithms

In this section, the relevant EOM are derived, containing the measurements and parameters of interest. Then these EOM are manipulated into forms that allow ID, first of the mass center and second of the inverse inertia matrix. Once the equations are in the proper form, either batch LS or RLS can be implemented.

3.1 Equations of motion

Starting with Euler's dynamical equation, and assuming the spacecraft inertia matrix is constant, the rotational EOM are [1]:

$$\dot{\omega} = I^{-1}(\tau - \omega \times (I\omega)), \text{ or, } \dot{\omega} = I^{-1}(\tau - \tilde{\omega} I\omega), \text{ or,}$$

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \begin{bmatrix} I_{11}^{-1} & I_{12}^{-1} & I_{13}^{-1} \\ I_{12}^{-1} & I_{22}^{-1} & I_{23}^{-1} \\ I_{13}^{-1} & I_{23}^{-1} & I_{33}^{-1} \end{bmatrix} \left(\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} - \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} I_{11} & I_{12} & I_{13} \\ I_{12} & I_{22} & I_{23} \\ I_{13} & I_{23} & I_{33} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \right)$$

where I is the spacecraft inertia matrix (and I_{ij}^{-1} indicates the (i, j) element of I^{-1} rather than $1/I_{ij}$), ω is the angular velocity of the body-fixed frame with respect to an inertial reference frame, τ is the sum of all torques on the body (thrusters, reaction wheels, CMGs, disturbances, *etc.*), and $\tilde{\omega}$ is used to represent the matrix-multiply implementation of the cross product.

The total torque, τ, about the true CM (which is not known exactly) due to the thrusters and torque disturbance is modeled as:

$$\tau = M_{thrusters} + \tau_{disturb} = (L \times D)F_k + \tau_{disturb}$$

where $M_{thrusters}$, represents the net thruster moment. L and D are [*3*-by-N] matrices (N is the number of thrusters) containing the location and direction (a unit-vector aligned with thrust vector) of each thruster, measured in the body frame. Each column contains the information for an individual thruster, and the cross product operator shown here indicates that the i^{th} column of L is crossed with the i^{th} column of D and entered into the i^{th} column of the resulting [*3*-by-N] matrix.

F_k, the thrust magnitude from each individual thruster at each control update, k, can be expressed as the following function, which accounts for (in order in the equation): reduction in thrust due to blowdown; nominal thruster magnitude; constant random bias added to the nominal value; random pulse-to-pulse offset added to the nominal value; thruster firing (on/off). A further refinement in calculating F_k would be to also model the transient nature of the thrusters, accounting for rise, trail off, and latency.

$$F_k = B(F_{nom} + F_{bias} + F_{random,k})T_k$$

B and F_{bias} can be identified separately. In this analysis, the ID algorithm uses their nominal values ($B = 1, F_{bias} = 0$), but their true random values are used in the simulation for testing.

Combining the above results, the full rotational EOM reduce to

$$\dot{\omega} = I^{-1}((L \times D)B(F_{nom} + F_{bias} + F_{random,k})T_k + \tau_{disturb} - \omega \times (I\omega))$$

The EOM are modeled assuming that the acceleration is constant during each control sample period; the ID algorithms require acceleration estimates corresponding to these control sample periods. The applications presented here estimate angular acceleration, $\dot{\omega}$, using rate gyros [13], but any method could be used (*e.g.*, using Kalman filtering, star trackers, accelerometers).

3.2 Identification of the mass center

The mass center, C (measured in the structural frame), determines the origin of the body frame, and thereby determines the value of L, which contains the locations of each thruster in the body frame. Similarly, Δ, the difference between actual and nominal values of C determines L. Δ is the value that will be identified here.

$$C \triangleq C_{nom} + \Delta, L = L_{nom} - \Delta[1 \quad 1 \quad \ldots \quad 1]$$

Recalling the rotational EOM, substituting and rearranging some terms, and setting $B = 1, F_{bias} = 0, F_{random,k} = 0, \tau_{disturb} = 0$,

$$\dot{\omega} = I^{-1}((L \times D)F_{nom}T_k - \omega \times (I\omega))$$

$$I\dot{\omega} + \omega \times (I\omega) = ((L_{nom} - \Delta[1 \quad 1 \quad \ldots \quad 1]) \times D)F_{nom}T_k$$

Trying to get this equation in the form $Ax \cong b$ with Δ as x,

$$I\dot{\omega} + \omega \times (I\omega) = (L_{nom} \times D)F_{nom}T_k - (\Delta[1 \quad 1 \quad \ldots \quad 1] \times D)F_{nom}T_k$$

Since the columns of $\Delta[1 \quad 1 \quad \ldots \quad 1]$ are all the same, the second term on the right can be regrouped. Note as a counter-example that the first term on the right cannot be similarly regrouped because the columns of L_{nom} are not all the same,

$$I\dot{\omega} + \omega \times (I\omega) = (L_{nom} \times D)F_{nom}T_k - (\Delta \times (DF_{nom}T_k))$$

$$\Delta \times (DF_{nom}T_k) = -I\dot{\omega} - \omega \times (I\omega) + (L_{nom} \times D)F_{nom}T_k$$

By the anti-commutative property of the cross product (that $a \times b = -(b \times a)$), the left side is changed to

$$(DF_{nom}T_k) \times \Delta = I\dot{\omega} + \omega \times (I\omega) - (L_{nom} \times D)F_{nom}T_k$$

The right side (the b in $Ax \cong b$) can be calculated. Introduce a variable, $c_k \equiv DF_{nom}T_k$ (a 3-by-1 vector), then re-write using the matrix-multiply implementation of the cross-product, as

$$\begin{bmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{bmatrix}_k \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} = I\dot{\omega} + \omega \times (I\omega) - (L_{nom} \times D)F_{nom}T_k$$

The measurements in this equation show up in the ω and $\dot{\omega}$ terms only. The $\omega \times (I\omega)$ term is very small (and can be omitted for the applications studied here), so $\dot{\omega}$ is the main noise contributor. The LS problem is defined (and closed-form solution developed) assuming the measurement noise enters without pre-multiplication. So, pre-multiplying by I^{-1},

$$I^{-1}\begin{bmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{bmatrix}_k \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} =$$

$$\dot{\omega} + I^{-1}(\omega \times (I\omega)) - I^{-1}(L_{nom} \times D)F_{nom}T_k$$

The $\omega \times (I\omega)$ term is still pre-multiplied, but that is likely to be an insignificant number. So at each update, $c_k \equiv DF_{nom}T_k$ is calculated and plugged in to the above re-formulation of the equation of motion, which conforms to $Ax \cong b$, where

$$A_k = I^{-1}\begin{bmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{bmatrix}_k, x = \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix},$$

$$b_k = \dot{\omega} + I^{-1}(\omega \times (I\omega)) - I^{-1}(L_{nom} \times D)F_{nom}T_k$$

3.3 Identification of the inverse inertia matrix

ID of I^{-1} is similar to that for Δ: the rotational EOM are manipulated to a form as close as possible to $Ax \cong b$. Recalling the original rotational EOM and simplifying as before,

$$\dot{\omega} = I^{-1}((L \times D)F_{nom}T_k - \omega \times (I\omega))$$

$$I^{-1}((L \times D)F_{nom}T_k - \omega \times (I\omega)) = \dot{\omega}$$

Introduce a variable, $a_k \equiv (L \times D)F_{nom}T_k - \omega \times (I\omega)$, so $I^{-1}a_k = \dot{\omega}$. The $I^{-1}a_k$ matrix-vector multiplication is converted into an equivalent expression where the 6 independent terms in I^{-1} appear in a vector that is pre-multiplied by a matrix.

$$I^{-1}a_k = \begin{bmatrix} I_{11}^{-1} & I_{12}^{-1} & I_{13}^{-1} \\ I_{12}^{-1} & I_{22}^{-1} & I_{23}^{-1} \\ I_{13}^{-1} & I_{23}^{-1} & I_{33}^{-1} \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_k = \begin{bmatrix} a_1 I_{11}^{-1} + a_2 I_{12}^{-1} + a_3 I_{13}^{-1} \\ a_1 I_{12}^{-1} + a_2 I_{22}^{-1} + a_3 I_{23}^{-1} \\ a_1 I_{13}^{-1} + a_2 I_{23}^{-1} + a_3 I_{33}^{-1} \end{bmatrix} \cdots$$

$$= \begin{bmatrix} a_1 & & & a_2 & a_3 & \\ & a_2 & & a_1 & & a_3 \\ & & a_3 & & a_1 & a_2 \end{bmatrix}\begin{bmatrix} I_{11}^{-1} \\ I_{22}^{-1} \\ I_{33}^{-1} \\ I_{12}^{-1} \\ I_{13}^{-1} \\ I_{23}^{-1} \end{bmatrix}$$

So now the equation is in the standard form, $Ax \cong b$, where

$$A_k = \begin{bmatrix} a_1 & & & a_2 & a_3 & \\ & a_2 & & a_1 & & a_3 \\ & & a_3 & & a_1 & a_2 \end{bmatrix}_k, x = \begin{bmatrix} I_{11}^{-1} \\ I_{22}^{-1} \\ I_{33}^{-1} \\ I_{12}^{-1} \\ I_{13}^{-1} \\ I_{23}^{-1} \end{bmatrix}, b_k = \dot{\omega}_k$$

For both mass-center ID and inertia ID, measurements at multiple time steps are combined as follows if the batch LS solution is used.

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix}$$

3.4 Deviations from standard LS ID form

The recursive or batch LS solution, $\hat{x} = (A^T A)^{-1} A^T b$, will truly minimize the quadratic error if the manipulated equations comply with the standard form, $Ax = b + \varepsilon$. Deviations from this form in the results of the preceding manipulations are that:

1. Noisy measurements appear in the $\omega \times (I\omega)$ term in the A matrix. However, for the relatively slow rotational speeds in many spacecraft applications, this term is negligible.
2. Other terms in A and b are not known perfectly: L, D, B, F_{bias}, *etc.* are all estimated or nominal values.
3. Random variables $F_{random,k}$ and $\tau_{disturb}$ that were set to zero do not appear directly in the ε term as they should.
4. CM ID uses nominal or estimated values for I and I^{-1}. Inertia ID uses nominal or estimated values for L (CM).

Due to the form of the underlying EOM, it is not possible to exactly comply with the standard form. The preceding manipulations and approximations to the EOM were chosen for this class of applications, attempting to minimize the expected effects of the inevitable resulting deviations. With different typical values of ω, disturbances, sensor accuracy, actuator variability, control policy, *etc.*, different formulations may be better (*e.g.*, [9]).

Also, the assumptions that acceleration is constant during each control update, and that thruster response time is zero will degrade the ID results if not compensated for.

4. Simulation Results

A dynamic simulation was developed using MATLAB [7] that included: the rotational and translational EOM; different sample rates for control and sensing; a thruster control system; and realistic variations in all relevant system parameters (mass properties, thruster properties and pulse-to-pulse noise, sensor noise).

The accuracy of the ID results depends on the sensor noise, thruster variability, and variability in non-identified system parameters (such as thruster direction, D, and bias, F_{bias}). These RLS mass-property ID algorithms have been implemented and successfully tested for the three dissimilar spacecraft mentioned. In this section, results for one, the Mini-AERCam, are presented as an example.

The Mini-AERCam has 12 thrusters controlling all 6 degrees of freedom, and uses MEMS gyros. In this test, the system was excited automatically by randomly perturbing the desired position and attitude, causing thrusters to fire, providing both torques and net forces on the vehicle.

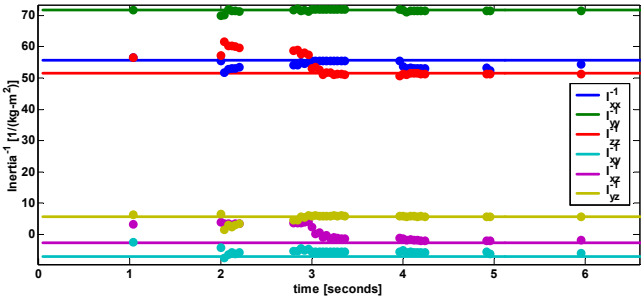

Fig 2: Mini-AERCam inertia-inverse ID results

Figure 2 shows the results for ID of the inverse inertia matrix. True values are drawn as solid lines; ID updates are drawn as dots. Since pure torques are common for this vehicle, inertia ID updates occur only when nominally pure torques are applied – resulting in the gaps between updates.

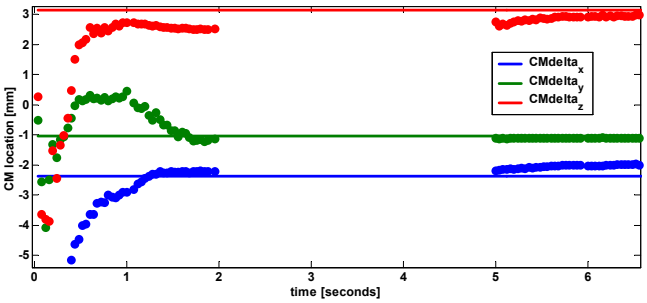

Fig 3: Mini-AERCam mass-center ID results

CM ID results are shown in Figure 3. As with inertia ID, the ID is initialized at the nominal values, which is zero in this case. CM information is not present when pure torques are applied, so there are no updates at these times (or when no thrusters fire), resulting in the gap between 2 and 5 seconds. In both inertia and CM ID, ID accuracy is better than the vehicle spec based on ground analysis and measurement (±5 mm for CM in this example), making the ID useful for on-line adaptive control.

Although they are run independently in these tests, use of CM ID results in the inertia ID calculations (*i.e.*, using identified instead of nominal CM information) and conversely would improve results.

5. Conclusions

Separate mass-center and inverse-inertia-matrix ID algorithms have been developed and applied in simulation to the X-38, Mini-AERCam, and S4 thruster-controlled vehicles. Using gyro signals only, and based on recursive least squares, the algorithms reliably and accurately ID mass properties for these vehicles in the presence of several significant noise sources. The algorithms are computationally efficient and can be run either on-line for adaptive control (RLS) or off-line for post-flight analysis (batch LS).

Acknowledgements

The authors wish to thank: Rodolfo Gonzalez, Dr. Steven Fredrickson, and Tim Straube of NASA Johnson Space Center and Dave Hammen of LinCom Corp. for providing valuable insights and information for the X-38 and Mini-AERCam applications; William Readdy and Gary Martin of NASA headquarters for programmatic support; and Richard Papasin, Alan Gasperini, and Rommel del Mundo of the NASA Ames SSRL for development of a 3-D visualization of the simulation.

References

[1] Chobotov, *Spacecraft Attitude Dynamics and Control*, 1991.

[2] Bergmann, E.V. and Dzielski, J., Spacecraft mass property identification with torque-generating control, *Journal of Guidance, Control, and Dynamics*, Vol. 13, Jan-Feb. 1990, p. 99-103.

[3] Bergmann, E.V., Walker, B.K., and Levy, D.R., Mass property estimation for control of asymmetrical satellites, *Journal of Guidance, Control, and Dynamics*, Vol. 10, Sept.-Oct. 1987, p. 483-491.

[4] Franklin, G., Powell, J.D., and Workman, M.L., *Digital Control of Dynamic Systems, Second Edition*, Addison Wesley Publishing, Menlo Park, California, 1990.

[5] Gelb, A, *et al*. *Applied Optimal Estimation*, MIT Press, Cambridge, Mass., 1974.

[6] Lawson, C. and Hanson, R., *Solving Least Squares Problems*, Series in Automatic Computation, Prentice Hall, Englewood Cliffs, NJ. 1974.

[7] MATLAB is a registered trademark of The MathWorks, Inc., 24 Prime Park Way, Natick, MA 01760, 508-647-7000.

[8] Richfield, R.F., Walker, B.K., and Bergmann, E.V., Input selection for a second-order mass property estimator, *Journal of Guidance, Control, and Dynamics*, Vol. 11, May-June 1988, p. 207-212.

[9] Tanygin, S. and Williams, T., Mass property estimation using coasting maneuvers, *Journal of Guidance, Control, and Dynamics*, Vol. 20, Jul-Aug 1997, p. 625-632.

[10] Willms, B., Space integrated GPS/INS (SIGI) navigation system for Space Shuttle. In *Digital Avionics Systems Conference Proceedings, 1999*.

[11] Wilson, E. and Rock, S.M., Reconfigurable control of a free-flying space robot using neural networks. In *Proceedings of the 1995 American Control Conference*, Seattle, Jun 1995.

[12] Wilson, E., *Experiments in Neural Network Control of a Free-Flying Space Robot*. PhD thesis, Stanford University, Stanford, CA 94305, March 1995.

[13] Wilson, E., Lages, C. R., and Mah, R.W., Gyro-based maximum-likelihood thruster fault detection and identification. In *Proceedings of the 2002 American Control Conference*, Anchorage, AK, May 2002.

[14] http://spaceflight.nasa.gov/station/assembly/elements /x38/